

# **Practical Algorithms for Image Analysis**

**DESCRIPTION, EXAMPLES, AND CODE**

**Michael Seul**

BioArray Solutions, LLC  
Piscataway, New Jersey

**Lawrence O'Gorman**

Veridicom, Inc.  
Chatham, New Jersey

**Michael J. Sammon**

Lucent Technologies  
Liberty Corner, New Jersey



PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE  
The Pitt Building, Trumpington Street, Cambridge, United Kingdom

CAMBRIDGE UNIVERSITY PRESS

The Edinburgh Building, Cambridge CB2 2RU, UK      <http://www.cup.cam.ac.uk>  
40 West 20th Street, New York, NY 10011-4211, USA      <http://www.cup.org>  
10 Stamford Road, Oakleigh, Melbourne 3166, Australia  
Ruiz de Alarcón 13, 28014 Madrid, Spain

© Cambridge University Press 2000, except computer programs,  
which are copyright by MLM SoftwareGroup

This book is in copyright. Subject to statutory exception  
and to the provisions of relevant collective licensing agreements,  
no reproduction of any part may take place without  
the written permission of Cambridge University Press.

First published 2000

Printed in the United States of America

Typeface Times Roman 10.5/13 pt.      System L<sup>A</sup>T<sub>E</sub>X 2<sub>E</sub> [TB]

*A catalog record for this book is available from  
the British Library.*

*Library of Congress Cataloging-in-Publication Data*

Seul, Michael.

Practical algorithms for image analysis : description, examples,  
and code / Michael Seul, Lawrence O'Gorman, Michael J. Sammon.

p.      cm.

1. Image processing – Digital techniques. 2. Algorithms. 3. Image  
analysis – Data processing. I. O'Gorman, Lawrence. II. Sammon,  
Michael J. III. Title.

TA1637.S38 1999

621.36'7 – dc21

98-18762

CIP

ISBN 0 521 66065 3 hardback

# Contents

<b>1</b>	<b>Introduction</b>	<i>page</i>	1
1.1	Introduction		1
1.2	Annotated Section Overview		5
1.3	A Guided Tour		14
<b>2</b>	<b>Global Image Analysis</b>		21
2.1	Intensity Histogram: Global Features		23
2.2	Histogram Transformations: Global Enhancement		27
2.3	Combining Images		37
2.4	Geometric Image Transformations		43
2.5	Color Image Transformations		51
<b>3</b>	<b>Gray-Scale Image Analysis</b>		56
3.1	Local Image Operations: Convolution		59
3.2	Noise Reduction		68
3.3	Edge Enhancement and Flat Fielding		75
3.4	Edge and Peak Point Detection		80
3.5	Advanced Edge Detection		90
3.6	Subsampling		97
3.7	Multiresolution Analysis		101
3.8	Template Matching		106
3.9	Binarization		110
<b>4</b>	<b>Binary Image Analysis</b>		118
4.1	Morphological and Cellular Processing		121
4.2	Binary Noise Removal		129
4.3	Region Detection		132
4.4	Shape Analysis: Geometrical Features and Moments		144
4.5	Advanced Shape Analysis: Fourier Descriptors		152
4.6	Convex Hull of Polygons		156
4.7	Thinning		160
4.8	Linewidth Determination		164
4.9	Global Features and Image Profiles		167
4.10	Hough Transform		172

<b>5 Analysis of Lines and Line Patterns</b>	176
5.1 Chain Coding	178
5.2 Line Features and Noise Reduction	184
5.3 Polygonalization	191
5.4 Critical Point Detection	197
5.5 Straight-Line Fitting	201
5.6 Cubic Spline Fitting	206
5.7 Morphology and Topology of Line Patterns	210
<b>6 Analysis of Point Patterns</b>	221
6.1 The Voronoi Diagram of Point Patterns	222
6.2 Spatial Statistics of Point Patterns: Distribution Functions	228
6.3 Topology and Geometry of Cellular Patterns	235
6.4 The $k$ -Nearest-Neighbor ( $k$ -NN) Problem	241
<b>7 Frequency Domain Analysis</b>	246
7.1 The 2D Discrete Fourier Transform	248
7.2 Frequency Domain Filtering	265
<b>APPENDIX: Synopsis of Important Concepts</b>	273
A.1 The Fourier Transform: Spatial Domain Versus Frequency Domain	274
A.2 Linear Systems: Impulse Response, Convolution, and Transfer Function	278
A.3 Special-Purpose Filters	283
A.4 The Whittaker–Shannon Sampling Theorem	285
A.5 Commonly Used Data Structures	288
<b>Index</b>	295

# 1 Introduction

## 1.1 Introduction

In this book, we offer guided access to a collection of algorithms for the digital manipulation of images. Our goal is to facilitate the solution of practical problems, addressing users whose interest is in an informed how-to approach, whether in a technical or in a casual setting. Rather than attempting to be exhaustive, we address salient practical considerations, guiding the selection of a particular approach to commonly encountered image processing and analysis tasks, and we present an implementation of our choice of the most suitable procedures. This selection of “Practical Algorithms” reflects our own experience as long-time users and developers of algorithms and software implementations to process and analyze images in areas as diverse as magnetic domain pattern formation and document analysis.

## HOW TO USE THIS BOOK

### **Organization of Chapters and Sections**

This book contains seven chapters and an appendix. Following this introduction, in Section 1.2, an annotated section overview is presented, and, in Section 1.3, a guide to the use of the book and the accompanying collection of algorithms are given. Chapters 2–7 present the material of the book in self-contained sections of identical format. The appendix serves as a review of fundamental concepts to which we refer throughout the text and provides reference material to the technical literature.

Each section contains a header that illustrates the nature of the topic of interest by describing typical applications, identifying key words, and providing cross references to related topics treated in other sections. Next, the topic of interest is introduced by a description of typical situations requiring a particular processing step or analytical operation; the effect of the operation is illustrated by a pictorial example that comprises a pair of before and after images. Possible strategies of implementation are then discussed, and a particular approach is selected for implementation. Annotated references provide an introduction to further technical literature. Appended to each section is a display of program usage for the code introduced in the section.

### **Single-Step Procedures**

Each section treats a single primary operation (histogram evaluation, low-pass filtering, edge detection, region detection, etc.) and introduces requisite algorithms. Each

## 2 Introduction

of the algorithms performs a single transformation on a given input image (“inimg”) to produce a modified output image (“outing”) and, in some cases, output data. Sections are self-contained to enable and encourage random access to the most suitable single-step procedure that solves the particular task of interest. Thus a reader interested in an exposition of simple edge detection techniques would open the book to Section 3.4, while a reader interested in the methodology of Fourier filtering would proceed directly to Section 7.2.

### Multistep Procedures

The analysis of images usually requires the successive application of multiple transformations. These may include simple preprocessing steps (noise removal, flat fielding, feature detection), followed by more complex analytical steps (object shape analysis, line pattern analysis, and point pattern analysis). Multiple individual transformations must be concatenated into multistep procedures.

To facilitate the flexible design of multistep procedures, sections are grouped into chapters that bundle common types of analyses (global, local, frequency domain) and common types of operations according to the images (gray scale, binary) or classes of patterns within images (lines, points) to which they are applied. Each section provides extensive cross references to enable a reader to construct a logical flow of related operations.

Chapters reflect the order in which procedures are ordinarily applied to any given image. For example, histogram analysis will precede binarization: filtering and/or flat fielding will precede object shape analysis, line coding will precede line pattern analysis, and Voronoi analysis will precede statistical analysis of point patterns. Spatial frequency analysis combines several operations in an alternative approach to the analysis of images in the spatial domain, and we choose to present it last. Within chapters, the order of sections reflects increasing levels of task complexity. This organization suggests an overall progression from simple and general to complex and specific tasks.

### Code

The book is accompanied by a collection of C programs implementing the algorithms we discuss. All programs operate on existing images; the acquisition of images is not discussed here. Versions for two platforms are supplied, as shown below.

#### Code Platforms

PLATFORM	C COMPILER
MicroSoft DOS (under Windows95)	MicroSoft Visual C++, v. 4.0
LINUX kernel v. 2.0.0	GNU gcc, v. 2.7.2

### Code Organization

Source code is organized in a directory structure such that modules required for building a particular program reside in a subdirectory corresponding to the book section in which the program is first invoked. For example, **xconv** would be stored in a

subdirectory referring to Section 3.1. Corresponding LINUX Makefiles and Visual C++ Workspaces are provided to generate executables. Details relating to program compilation and installation are described in a README.TXT file contained on the code distribution disk. In some chapters, additional utilities are provided in a separate subdirectory to generate test images or to analyze output data files.

In addition, we provide three libraries:

- **LIBTIFF**, containing functions to handle input and output of image files in an uncompressed Tagged Image File (TIF) format; *LIBTIFF* 3.4, written by Sam Leffler and made available by Silicon Graphics, Inc., was modified to remove support for LZW compression;
- **LIBIMAGE**, containing `getopt()`, a command line parser, as well as a collection of higher-level graphics functions to handle drawing, filling, and character generation; portions of *LIBIMAGE* invoke *gd* 1.2, written by Tom Boutell and made available by Quest Protein Database Center, Cold Spring Harbor Labs;
- **LIBIP**, containing image analysis functions such as `poly_moments()` and `find_area_hist()` to handle common analytical tasks invoked by multiple programs.

All programs are executed from the command line from which a variety of arguments and options can be supplied to adjust and optimize program performance. Some programs prompt for additional input during run time. When executed without arguments, each program displays the command with arguments and options defining program usage. This usage header also is reproduced at the end of each section for programs introduced in that section.

### **Notices**

This source code is distributed under a limited-use license that one may view by invoking an appropriate option on the command line of each program. Unless otherwise indicated in the source code, copyright is jointly held by the authors, as indicated by a copyright notice such as Copyright (C) 1997, 1998, 1999, M. Seul, L. O’Gorman, M. J. Sammon. In some instances, we use a third-party code in versions available from public sources. In those instances, the original authors, as identified in the source code, retain copyright.

Except when otherwise stated in writing, the copyright holders and/or other parties provide each program as is without any warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for particular purpose. The entire risk as to the quality and performance of the programs is with the user. Should the program prove defective, the user assumes the cost of all necessary servicing, repair, or correction.

In no event shall any copyright holder be liable for damages including any direct, indirect, general, special, incidental, or consequential damages arising out of the use or inability to use the programs (including, but not limited to, loss of use or data, data being rendered inaccurate, losses sustained by the user or third parties, or a failure

## **4      Introduction**

of the programs to operate with any other programs), however caused, and under any theory of liability, whether in contract, strict liability, or tort.

### **Acknowledgments**

We would like to thank those individuals, most notably our editor, Lauren Cowles, who have contributed to this book. This publication includes images from Corel Stock Photo Library, which are protected by the copyright laws of the United States, Canada, and elsewhere and are used under license. Images and other printed materials have been reproduced with permission from the following sources, listed in the order of first appearance and further identified in the referenced sections: [O’Gorman et al 85] – Figs. 2.1.1, 2.2.2, 2.2.5, 5.2.4, 5.3.1, 5.6.1; [CSPL – Corel Stock Photo Library/Corel Group] – Figs. 2.2.1, 2.2.6, 2.5.1, 3.6.1, 3.7.1, 4.9.2, 5.5.1, 7.1.6, 7.1.7; [Morgan and Seul 95] – Figs. 2.3.1, 3.3.1; [Woodward and Zasadzinski 96] – Fig. 2.4.1; [The Bell Labs Technical Journal] – Fig. 2.4.6; [Costello] – Figs. 3.1.1 B, 7.1.1; [Gonzalez and Woods 92] – Figs. 3.1.3, 3.1.4, 3.4.2; [Lucent Technologies – Bell Laboratories Physical Sciences Image Library] – Figs. 3.2.1, 3.2.4, 4.3.1; [Seul et al 92] – Figs. 3.3.2, 5.3.2, 5.7.2–5; [Canny 86] – Fig. 3.5.2; [Boie et al 86] – Figs. 3.5.3, 3.5.4; [Seul et al 91] – Figs. 4.3.2, 4.3.3, 4.5.2, 4.5.3; [Attneave 54] – Fig. 5.4.2; [Seul and Wolfe 92B] – Fig. 5.7.1; [Seul and Wolfe 92A] – Fig. 5.7.6; [Preparata and Shamos 85] – Figs. 6.1.2., 6.1.4; [Ashcroft and Mermin 76] – Fig. 6.1.2; [Fortune 87] – Fig. 6.1.5; [Sire and Seul 95] – Figs. 6.3.1-3, 6.4.1; [Seul and Chen 93] – Fig. 7.1.8; [Goodman 68] – Figs. 7.2.1, A.3; [Bracewell 86] – Figs. A.1, A.2; [Sessions 89] – Fig. A.4; [Kruse et al 91] – Fig. A.5.

## 1.2 Annotated Section Overview

### Chapter 1. Introduction

SECTION TITLE	SYNOPSIS
1.1 Introduction	Describes organization of chapters and sections and code
1.2 Annotated Section Overview	Functions as table of contents (section titles) with annotation in the form of section headers (Typical Application(s), Key Words, Related Topics)
1.3 A Guided Tour	Illustrates use of code to construct multistep sequences of image processing and analysis operations

### Chapter 2. GlobalImageAnalysis

SECTION TITLE	TYPICAL APPLICATION	KEY WORDS	RELATED TOPICS <sup>†</sup>
2.1 Intensity Histogram: Global Features	image quality test, object location	intensity histogram, global image features	global enhancement (2.2), binarization (3.9)
2.2 Histogram Transformation: Global Enhancement	enhancement of image contrast	histogram transformations, expansion, equalization; mapping function	global features (2.1), local operations (3.1)
2.3 Combining Images	image overlays, background subtraction; XOR binary operation to control composite image	addition, subtraction; AND, OR, XOR	histogram operations (2.1), color image transformations (2.5), flat fielding (3.3), binarization (3.9), morphological and cellular processing (4.1)
2.4 Geometric Image Transformations	scaling; rotation	interpolation, resampling; image magnification and reduction; rotation	geometric image transformations (2.4), subsampling (3.6), multiresolution analysis (3.7), morphology and cellular processing (4.1), sampling (A.4)
2.5 Color Image Transformations	color, bases transformations, intensity, hue, saturation, RGB, IHS, YIQ	gray-scale analysis (Chap. 3)	

<sup>†</sup> The numbers in parentheses indicate the section in which these topics can be found.

## Chapter 3. Gray-Scale Image Analysis

SECTION TITLE	TYPICAL APPLICATION	KEY WORDS	RELATED TOPICS
3.1 Local Image Operations: Convolution	image contrast manipulation by means of filtering, required in many of the applications discussed in subsequent sections	filter mask, kernel; smoothing, sharpening; edge and point detection; transfer function; cyclic convolution	smoothing/noise reduction (3.2), feature enhancement (3.3), edge detection (3.4), filter mask design (7.2), correlation (A.1)
3.2 Noise Reduction	reduction of extraneous image features; reduction of noise introduced by imaging system	noise removal, filtering, low-pass filter, median filter, speckle noise, smoothing, blurring	subsampling (3.6), binary noise removal (4.2), line noise reduction (5.2), frequency domain filtering (7.2)
3.3 Edge Enhancement and Flat Fielding	emphasis of localized features such as contours; elimination of inhomogeneities in scene illumination	high-pass filter, sharpening; unsharp mask, flat fielding	convolution (3.1), edge detection (3.4); shape analysis (4.4), Hough transform (4.10); frequency domain filtering (7.2)
3.4 Edge and Peak Point Detection	detection of lines and points; locate intensity peaks of extended pointlike objects; image segmentation	gradient, Laplacian; region peak detection, converging squares	edge enhancement (3.3), advanced edge detection (3.5), multiresolution analysis (3.7), template matching (3.8); binary region detection (4.3), shape analysis (4.4), Hough transform (4.10)

3.5 Advanced Edge Detection	image segmentation in the presence of noise	optimal detection, matched filter, Wiener filter	edge enhancement (3.3), edge detection (3.4), multiresolution analysis (3.7), template matching (3.8), shape analysis (4.4), Hough transform (4.10)
3.6 Subsampling	scale reduction of images to be subjected to object or feature detection	subsampling, image size reduction, resolution adjustment	geometric interpolation (2.4); noise reduction (3.2), multiresolution analysis (3.7); frequency domain filtering (7.2)
3.7 Multiresolution Analysis	object detection	scale-space processing, multiresolution analysis, multiresolution pyramids	noise reduction (3.2), edge enhancement (3.3), subsampling (3.6), template matching (3.8)
3.8 Template Matching	detection of objects of known shape in noisy environment	matched filtering, template matching, cross correlation	binary region detection (4.3), shape analysis (4.4), spectral shape analysis (4.5), Hough transform (4.10); critical point detection (5.4)
3.9 Binarization	conversion of gray-scale to binary image	thresholding, local, global, contextual	image intensity histogram (2.1), histogram transformations (2.2), edge detection (3.4)

## Chapter 4. Binary Image Analysis

SECTION TITLE	TYPICAL APPLICATION	KEY WORDS	RELATED TOPICS
4.1 Morphological and Cellular Processing	modification of region shapes in binary images	mathematical morphology, cellular logic; erosion, dilation, shrink, expand; region growing; structuring element; binary filtering	template matching (3.8), binary noise reduction (4.2), thinning (4.7), linewidth determination (4.8)
4.2 Binary Noise Removal	reduction of noise in binary images	“salt-and-pepper” noise; <i>k</i> Fill	noise reduction (3.2)
4.3 Region Detection	segmentation of images by delineation of regions and encoding of contours	region detection, segmentation, contour representation, cumulative angular bend, curvature point, region filling or coloring, connected component labeling; contour detection, polygonal representation, tangential and radial contour representation	peak detection (3.4); object shape analysis (4.4), thinning (4.7)
4.4 Shape Analysis: Geometrical Features and Moments	shape descriptors for objects, particularly those of near-circular shape, e.g., phospholipid vesicles and erythrocytes, and for domains formed in a wide variety of physical-chemical systems	global shape descriptor, curvature energy, moments, moments of inertia, moment invariants, recursive evaluation	edge and peak detection (3.4); spectral shape analysis (4.5), convex hull (4.6); polygonalization (5.3), critical point detection (5.4)

4.5	Advanced Shape Analysis: Fourier Descriptors	analysis of object shape, based either on object area or on object contour	shape analysis, spectral shape analysis, Fourier descriptors	region growing (4.3), shape descriptors (4.4)
4.6	Convex Hull of Polygons	delineation of polygonal region; association of a shape with a group of points forming the vertices of a polygon	extreme points, convex hull, shape	shape analysis (4.4), (4.5)
4.7	Thinning	thinning ("skeletonization") of elongated regions, lines, and contours	skeleton, medial axis transform	polygonalization (5.3), line fitting(5.5)
4.8	Linewidth Determination	thinning of an image with simultaneous retention of linewidth information	augmented thinning, line image reconstruction	thinning (4.7)
4.9	Global Features and Image Profiles	global analysis of collections of multiple objects	statistical features, image moments, image projection profiles, intensity signatures	global image features (2.1), multiresolution analysis (3.7), shape analysis (4.4), two-dimensional Fourier transform (7.1)
4.10	Hough Transform	detection of lines (to a lesser degree, of other shapes such as circles) in noisy images	Hough transform, line fitting	template matching (3.8); shape features (4.4); line fitting (5.5); two-dimensional Fourier transform (7.1)

## Chapter 5. Analysis of Lines and Line Patterns

SECTION TITLE	TYPICAL APPLICATION	KEY WORDS	RELATED TOPICS
5.1 Chain Coding	efficient representation of line patterns such as contour maps, engineering diagrams, fingerprints, and magnetic domain patterns	directional coding, chain code	region detection (4.3), thinning (4.7)
5.2 Line Features and Noise Reduction	recording of line pattern features such as branch and end points; removal of spurious line features from thinned patterns	matched line filters, chain code, primitives chain code (PCC), thin line code (TLC)	noise reduction (3.2); binary noise removal (4.2), thinning (4.7); chain code (5.1)
5.3 Polygonalization	smoothing and parameterization of noisy contours or edges	straight-line approximation, curve representation	thinning (4.7); chain coding (5.1), critical point detection (5.4), line fitting (5.5)
5.4 Critical Point Detection	curve shape description, identification of curvature maxima along contours	critical points, dominant points, curvature, curvature plot, curvature maxima and minima, corner detection, difference of slopes (DoS), $k$ -curvature	shape features (4.4); polygonalization (5.3), line fitting (5.5)

5.5 Straight-Line Fitting	detection and parameterization of straight lines, especially in diagrams	line fitting, straight-line fitting, least-squares fit, regression fit, eigenvector line fitting, principal-axis line fitting	Hough transform (4.10); polygonalization (5.3), critical point detection (5.4)
5.6 Cubic Spline Fitting	approximation of curves and contours by smooth polynomial, noise reduction, smoothing	spline fit, B-splines, cardinal splines, approximating splines, interpolating splines, third-order polynomial fit	polygonalization (5.3), critical point detection (5.4), line fitting (5.5)
5.7 Morphology and Topology of Line Patterns	direct-space analysis of line patterns to ascertain local ordering and to derive a quantitative description of morphological determinants; identification of topological (point) defects	parallelism, overlap, and adjacency; segment clusters; cluster geometry and global descriptors; line pattern topology: branch and end points, disclinations	convex hull (4.6), thinning (4.7); chain coding (PCC) (5.1), noise reduction (TLC) (5.2), polygonalization (5.3)

## Chapter 6. Analysis of Point Patterns

SECTION TITLE	TYPICAL APPLICATION	KEY WORDS	RELATED TOPICS
6.1 The Voronoi Diagram of Point Patterns	tesselation of planar patterns of point particles, such as those in images of atoms, molecules, or cells adsorbed to surfaces, layers of colloidal spheres in suspension, domain patterns in a wide variety of physical–chemical systems or stars and galaxies	planar graph, tesselation, triangulation, proximity, nearest neighbor (NN)	region detection (4.3), medial-axis transform (thinning) (4.7), point pattern analysis (Chap. 6)
6.2 Spatial Statistics of Point Patterns:	evaluation of distribution functions for pair distances and coordination numbers to assess degree of randomness	distance statistics, angle statistics; random, clustered, ordered patterns	point detection (3.4),
Distribution Functions	statistics of cellular patterns and polygonal networks; analysis of pattern coarsening dynamics	coordination number, topological charge, Lewis law, Aboav–Weaire law	$k$ -NN shells (6.4)
6.3 Topology and Geometry of Cellular Patterns	partitioning of a given point set into clusters; range finding, particle tracking; evaluation of pattern statistics as a function of increasing index, $k$	charge compensation	nearest-neighbor (NN) shell, fractal measure, range finding, clustering
6.4 The $k$ -Nearest-Neighbor ( $k$ -NN) Problem			proximity problem (6.1), cellular patterns (6.3)

## Chapter 7. Frequency Domain Analysis

SECTION TITLE	TYPICAL APPLICATION	KEY WORDS	RELATED TOPICS
7.1 The Two-Dimensional Discrete Fourier Transform	segmentation and recognition by global pattern or texture; evaluation of diffraction patterns; filtering; convolution	discrete Fourier transform (DFT), fast Fourier transform (FFT); inverse Fourier transform; correlation, power spectrum; sampling rate, resolution, maximum frequency, minimum size; aliasing, “jaggies”; windowing	convolution (3.1), subsampling (3.6), multiresolution analysis (3.7); Fourier descriptors (4.5)
7.2 Frequency Domain Filtering	smoothing, edge detection, texture segmentation, pattern segmentation	filtering: low-pass, high-pass, bandpass, band-stop; cutoff frequency; Gaussian filter, Butterworth filter	convolution (3.1), noise reduction (3.2), edge enhancement (3.3), subsampling (3.6), multiresolution analysis (3.7)

## Appendix. Synopsis of Important Concepts

SECTION TITLE	INCLUDED TOPICS
A.1 The Fourier Transform: Interconversion between Spatial Domain and Frequency Domain	properties of the Fourier transform
A.2 Linear Systems: Impulse Response, Convolution, and Transfer Function	convolution, impulse response, transfer function
A.3 Special-Purpose Filters	matched filter, Wiener filter
A.4 The Whittaker–Shannon Sampling Theorem	sampling and reconstruction of band-limited functions
A.5 Commonly Used Data Structures	lists and trees

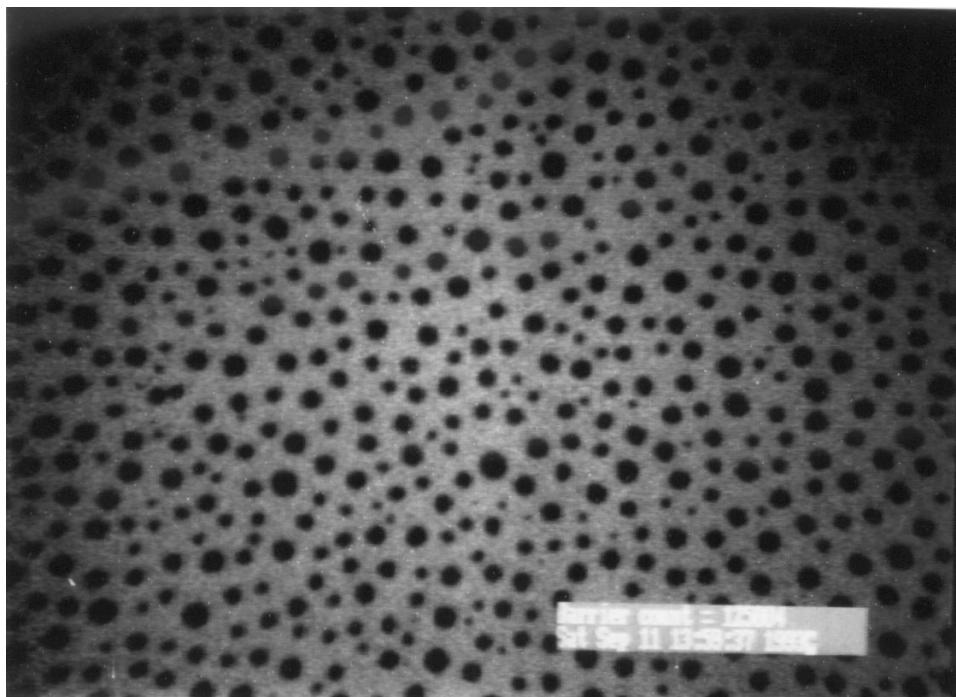
### 1.3 A Guided Tour

Programs in each section implementing single transformations are self-contained to enable the concatenation of multiple single-step operations. As illustrated in the Guided Tour below, readers are encouraged to compose their own sequences of operations, stepping from section to section by following cross references and proceeding in the general direction from lower to higher chapters.

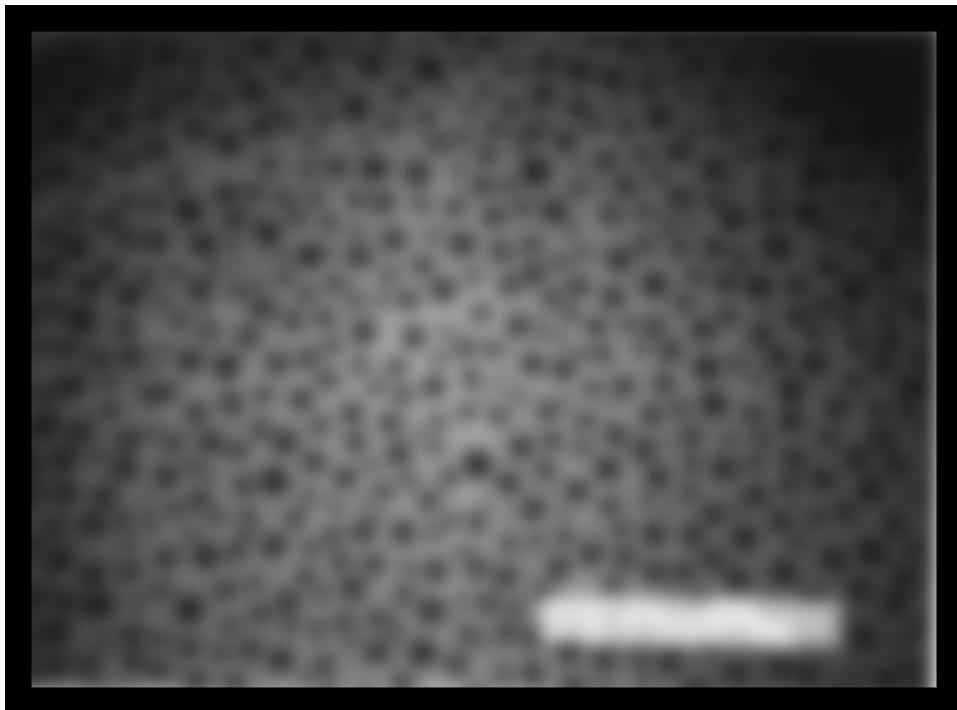
Applied to the original image, the set of operations listed below produces the series of images shown in this section. We describe each step in turn with reference to specific sections in the book and reproduce a corresponding command line (with comments).

**Original** The starting point for our tour is a raw image in the form of a micrograph depicting a set of small, roughly circular dark domains embedded in a bright background (Fig. 1.3.1). These domain patterns form in certain organic films floating on water, similar to soap films; different regions within the films are made visible by fluorescent dye molecules that avoid the interiors of the domains, which therefore remain dark; contrast in these images tends to be low.

**Reference Image** The first processing step is to remove the significant spatial nonuniformities in the illumination profile that would otherwise introduce bias into the intensity histogram and interfere with subsequent global processing steps such as binarization. Nonuniformities in scene illumination are the rule, not the exception: for example, microscope illuminators are notoriously nonuniform. The requisite correction requires a reference image showing the slow background variations (but not the objects of interest). Unless otherwise available, for example, in the form of a separately stored image of the scene background, a reference image is created from the original by



**Figure 1.3.1.**



**Figure 1.3.2.**

low-pass filtering; **xconv** (Section 3.1), used with a Gaussian low-pass filter (Section 3.2), produced Fig. 1.3.2.

---

```
xconv domains.tif domREF.tif -g 65
REM Section 3.1
REM     Generate 2D Gaussian of size 65x65
REM     Produce reference image ("background") by low-pass filtering
REM     Store output in domREF.tif
REM Note: REM indicates comment line
```

---

**Background Correction** With reference image in hand, we proceed to implement the actual background correction by applying an operation known as flat fielding. This involves the division of the original by the reference image (Section 3.3) and the subsequent scaling of resulting intensities to enhance contrast; **bc** (Section 3.3) produced Fig. 1.3.3.

---

```
bc domains.tif domREF.tif domBC.tif
REM Section 2.3
REM     Correct background nonuniformities: divide by reference image;
REM     Scale intensities
REM     Store output in domBC.tif
```

---

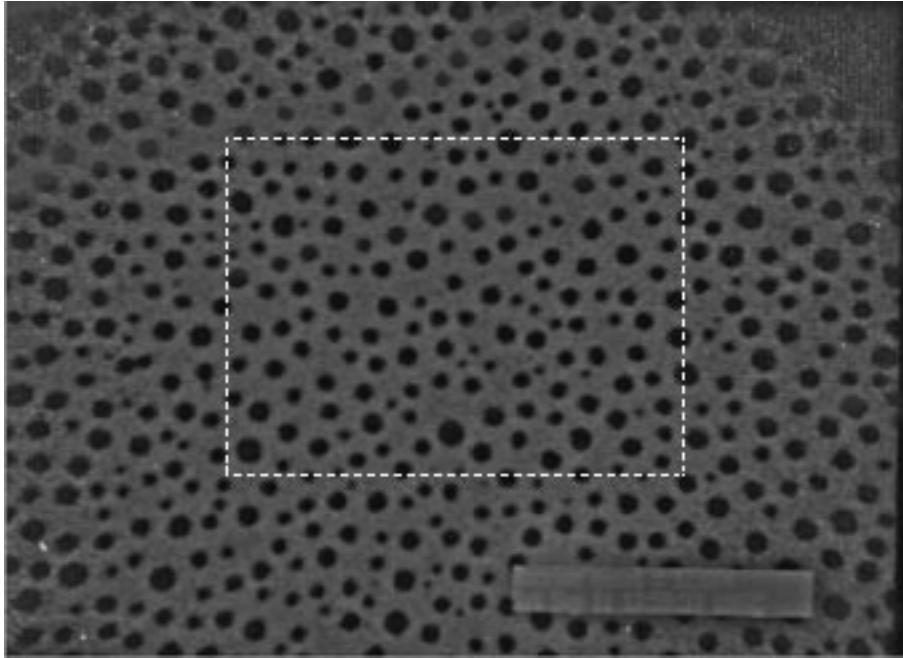


Figure 1.3.3.

**Binarization** Binarization represents the final step in processing the image to convert it into a form that is suitable for subsequent analysis of image content. `threshm`, an optimal thresholding routine (Section 3.9), was applied to a portion of the image, also referred to as area of interest (AOI); this is selected by supplying AOI boundaries as command line arguments. Binarization was followed by magnification (Section 2.4) by means of `xscale` to produce Fig. 1.3.4.

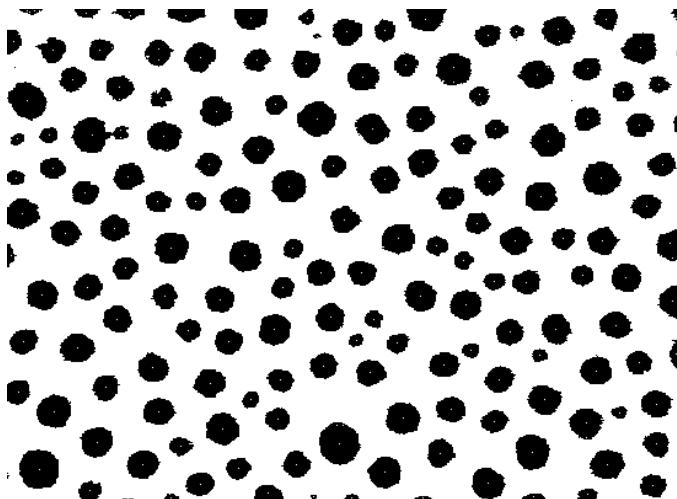


Figure 1.3.4.

---

```

threshm domBC.tif domBIN.tif
REM Section 3.9
REM Specify A(rea)O(f)I(nterest)?y
REM Input upper left, lower right AOI coordinates 300 200 850
REM 600
REM Calculated threshold (by moment method) = 63.
REM Note: only a portion of the input image, specified by AOI,
REM is binarized
REM           the binarized image appears magnified

```

---

**Optional: Edge Detection** As an optional step, domain edges in the binarized image are readily identified and marked by applying **xconv** in conjunction with a  $3 \times 3$  Laplacian filter mask; the image is not displayed here. More typically, simple (Section 3.4) or advanced (Section 3.5) edge detection would be performed before binarization.

---

```

xconv domBIN.tif domLAP.tif -f laplac33.txt
REM Section 3.4
REM Perform edge detection by using 3 x 3 Laplacian, provided in
REM laplac33.txt
REM      0.00000 -1.00000  0.00000
REM     -1.00000  4.00000 -1.00000
REM      0.00000 -1.00000  0.00000
REM Doing image convolution...
REM NOTE: for demonstration only;
REM       edge detection usually applied to gray-scale image

```

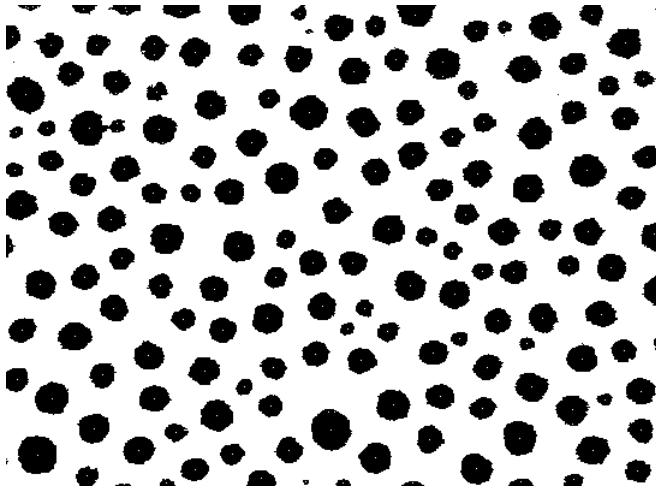
---

**Analysis** A variety of analytical steps may now be applied to the binarized image to extract quantitative information regarding image content. The objective in analyzing this type of image is twofold:

- analyze the shape of individual domains in order to measure deviations from circularity and to ascertain the balance of underlying forces,
- analyze positions of domains within the pattern to ascertain the presence or absence of spatial correlations: for example, is the pattern of domains completely random or does it reveal any regularity?

**Domain Contours: Shape Descriptors** First, in pursuing the detailed analysis of individual domain shapes, we may wish to determine suitable shape descriptors to reveal the degree of distortion of domains away from a circular reference state (Sections 4.4 and 4.5).

We may accomplish this, for example, by invoking **xcp** (Section 4.3), which performs region encoding, or by invoking **xpm** (Section 4.4), which combines region encoding with contour analysis in terms of geometrical shape descriptors and moments; or, as shown here, by invoking **xbdy** (Section 4.5), which goes beyond **xpm** to provide a spectral representation of shape features in terms of Fourier descriptors.

**Figure 1.3.5.**


---

```

xbdy domBIN.tif domBDY.tif -w domBIN.acm
REM Sections 4.4, 4.5
REM      Scan, link edge points, encode contour(s);
REM      Select domain(s) for shape analysis;
REM      Select various options (in response to prompting);
REM      Evaluate various shape descriptors;
REM      Create output image, containing:
REM          resampled contour/edge points; centroid, principal axes;
REM      Write data to domBIN.acm file;
REM      NOTE: output image (domBDY.tif) not shown
REM          use XOR (see IMGBOOL, sect. 2.3) to overlay onto
REM          original

```

---

**Domain Pattern: Region Filling, Centroids** An alternative mode of analysis would be to focus on the spatial configuration of domains within the pattern rather than on individual domain shapes. Accordingly, to locate domains in the pattern, an AOI within the image is scanned. The AOI is identified by means of coordinates supplied as command line arguments to **xah** (Section 4.3), a routine that scans the pattern, fills regions (Section 4.3), and produces a point pattern composed of domain centroids; this is shown superimposed on a binarized domain pattern in Fig. 1.3.5, a figure produced by addition (Section 2.3) of the point pattern and the binarized images.

---

```

xah domBIN.tif domAH.tif
REM Sections 4.3, 4.5
REM      Scan, fill regions, and generate point pattern of centroids;
REM      Produce area histogram;

```

---